

Ontologia para o mapeamento semântico entre ambientes virtuais de aprendizagem

Danilo G. Carlos¹, Ademar F. de Sousa Neto², Francisco M. Mendes Neto¹,
Bruno de S. Monteiro¹, Salatiel D. Silva¹, Igor F. Santos¹

¹Programa de Pós-Graduação em Ciência da Computação
da Universidade Federal Rural do Semi-Árido (UFERSA)

²Programa de Pós-Graduação em Ciência da Computação
da Universidade Federal de Campina Grande (UFCG)

{danilocarlos, ademarneto, miltonmendes}@ufersa.edu.br,

{brunomonteiro, salatiel.dantas, igorsantos}@ufersa.edu.br

Abstract. *Integrating heterogeneous systems requires data to be mapped in a format that can be understood by the systems involved in communication. Using ontology allows for more flexibility and semantic mapping of data. Thus, this paper aims to present the development and evaluation of an ontology for semantic mapping between virtual learning environments called LESD-Onto. An ontology was built based on the Ontology Development 101 methodology and the aspects of consistency, conclusion and conciseness. For evaluation, an ontology was used in the integration of Edubi and Moodle systems, the allowed results indicate its viability in the integration of learning systems.*

Resumo. *Integrar sistemas heterogêneos exige que os dados sejam mapeados em um formato que possa ser compreendido pelos sistemas envolvidos na comunicação. O uso de ontologia permite mais flexibilidade e o mapeamento semântico dos dados. Desse modo, este trabalho tem como objetivo apresentar o desenvolvimento e a avaliação de uma ontologia para mapeamento semântico entre ambientes virtuais de aprendizagem chamada LESD-Onto. A ontologia foi construída com base na metodologia Ontology Development 101 e nos aspectos de consistência, completude e concisão. Para avaliação, a ontologia foi utilizada na integração dos sistemas Edubi e Moodle, cujos resultados dão indícios de sua viabilidade na integração de sistemas de aprendizagem.*

1. Introdução

A popularização das tecnologias digitais móveis, a valorização dos ambientes informais nas práticas educacionais e a sensibilidade ao contexto dos indivíduos viabilizam o desenvolvimento de práticas ainda mais flexíveis, favorecendo a aprendizagem em qualquer lugar, a qualquer momento, adequada às características do indivíduo e independente do tipo de dispositivo digital disponível [Romaní and Moravec 2011].

Neste sentido, a aprendizagem ubíqua (*ubiquitous learning* ou *u-learning*) surge da combinação de modelos de aprendizagem flexíveis e tecnologias digitais com o objetivo de prover ferramentas que ampliam as oportunidades de práticas de ensino e aprendizagem. Ferramentas baseadas nesse conceito permitem oferecer aos aprendizes e pro-

fessores conteúdos e serviços relevantes com base no contexto no qual os sujeitos estão inseridos [de Sousa Monteiro et al. 2015].

Contudo, apesar das ferramentas de *u-learning* se apresentarem como soluções inovadoras, algumas características dos ambientes de aprendizagem tradicionais, especialmente os sistemas de gestão da aprendizagem (SGA), também chamados de *Learning Management Systems* (LMS), ainda são fundamentais para as instituições que ofertam cursos na modalidade de educação a distância (EAD). Portanto, esta pesquisa adota a perspectiva de que os ambientes de *u-learning* complementam a experiência dos alunos, dos tutores e dos professores usuários de outros ambientes de aprendizagem, de modo a contribuir com a adoção de novas formas de interação e práticas de ensino-aprendizagem.

Existem diversas estratégias que podem ser utilizadas para integrar aplicações, tais como utilizar uma base de dados compartilhada, transferir arquivos, realizar chamadas para procedimentos e transferir mensagens [Hohpe and Woolf 2004]. Contudo, com a ascensão da web, diversas aplicações vêm adotando os *web services* para prover serviços às aplicações externas. Eles fornecem interoperabilidade tanto por meio de chamadas para procedimentos remotos como pelo envio e recebimento de mensagens aproveitando a estrutura da web [Vinoski 2002].

Uma das principais vantagens da utilização de *web services* é a possibilidade de integrar aplicações construídas em diferentes plataformas e linguagens de programação, permitindo que aplicações heterogêneas possam ser integradas. Isso ocorre porque são adotados padrões e protocolos onipresentes como o XML e o HTTP, amplamente utilizados em sistemas *Web*. Nos sistemas de software voltados para educação, diversas plataformas utilizam *web services* para prover integração com outras aplicações, tais como: Moodle, OpenRedu, Amadeus LMS e Youubi [de Sousa Monteiro et al. 2015].

Mesmo com a adoção de *web services*, oferecer uma visão uniforme dos dados que transitam entre sistemas heterogêneos não é trivial, pois os dados disponibilizados pelos serviços precisam ser mapeados para um formato compreensível pelas aplicações envolvidas na comunicação. Esse mapeamento pode ser realizado de maneira sintática direto no código fonte, o que aumenta o grau de acoplamento. Esse processo demanda maior esforço ao desenvolvedor para integrar e manter o software funcionando, visto que, ocorrendo uma alteração no *web service*, é preciso alterar o código-fonte de todos os sistemas que consomem seus serviços. Além disso, os modelos de dados de ambientes heterogêneos podem conter informações semanticamente diferentes, ou seja, o significado de um conceito em um ambiente pode ser interpretado de maneira diferente no outro ambiente. Entretanto, a utilização de ontologias na integração de sistemas pode contornar conflitos desse tipo.

No contexto da Ciência da Computação, uma ontologia permite definir primitivas de representação que podem modelar um determinado domínio de conhecimento. Essas primitivas podem ser classes, atributos e relacionamentos que podem atribuir significado a um determinado conceito. As ontologias também permitem adicionar um nível de abstração de modelos de dados, com a intenção de modelar o conhecimento sobre indivíduos, seus atributos e relacionamentos [Gruber et al. 2009]. Devido sua flexibilidade, ontologias podem ser utilizadas, por exemplo, no mapeamento e integração de bancos

de dados heterogêneos, tornando possível realizar a exportação de dados entre diferentes sistemas. A ontologia pode atuar como um dicionário que descreve o modelo de dados de cada ambiente de aprendizagem e a relação entre eles, mapeando seus conceitos para um vocabulário comum.

Tendo em vista essa problemática, o presente trabalho tem como objetivo apresentar uma ontologia para o mapeamento semântico de dados em ambientes de aprendizagem. A ontologia será utilizada para identificar como os serviços e dados providos por serviços de diferentes ambientes de aprendizagem podem ser representados. O uso de ontologia foi escolhido devido a sua flexibilidade, simplicidade no mapeamento e manutenção dos conceitos.

Para construir a ontologia proposta, foi utilizada a *Ontology Development 101* [Noy et al. 2004]. Para validar a ontologia, foram avaliados os aspectos de consistência, completude e concisão, conforme sugerido em Gómez-Pérez (2004). A ontologia será encapsulada em um componente “Integrador de LMS” (Figura1), que fornece ao ambiente Edubi a integração com outros ambientes de aprendizagem. No escopo deste trabalho, optou-se pela integração com o ambiente Moodle, por ser bastante difundido e por ser utilizado nos cursos da Universidade Aberta do Brasil. No entanto, a ontologia foi projetada para ser expandida para outras plataformas EaD e *Blended Learning*, tais como AmadeusLMS, OpenRedu, entre outras.

O presente trabalho está organizado da seguinte forma: a Seção 2 apresenta o Edubi e suas características; a Seção 3 apresenta a metodologia utilizada na construção da ontologia; na Seção 4 é descrita a ontologia proposta e como ela foi modelada; a Seção 5 apresenta o modelo de avaliação da ontologia e os resultados obtidos; por fim, a Seção 6 apresenta as considerações finais e trabalhos futuros.

2. O Ambiente de Aprendizagem Ubíqua Edubi

O ambiente Edubi, representado na Figura 1, é composto por quatro aplicações clientes (*Web*, *Mobile*, *Watch* e *TV*), cada uma delas com suas particularidades, e dois componentes de integração, onde um deles possibilita a essas aplicações consumirem os serviços disponibilizados pelo Moodle, e o outro descrito em Bezerra (2017), permite a recomendação de conteúdos da *Wikipedia* e *Youtube*. Além desses, os artefatos de software do ambiente Edubi consomem também os serviços da Youubi API [de Sousa Monteiro et al. 2015], componente central da arquitetura.

O processo de design do Edubi priorizou cenários de educação a distância, porém, pode ser adotado também nas modalidades presencial e semipresencial, dentro do paradigma da aprendizagem ubíqua e informal. Para isso, faz uso de funcionalidades cujos serviços base são providos pela Youubi API.

O Youubi [de Sousa Monteiro et al. 2015] é uma API (*Application Programming Interface*), desenvolvida com a linguagem Java, que fornece serviços que auxiliam o desenvolvimento de aplicações clientes voltadas à computação ubíqua, mas que pode ser adotada em diversos domínios específicos (educação, saúde, entretenimento, turismo, comunicação, marketing, etc). Ela permite a criação e a manipulação de sete Entidades Elementares (EE): pessoa, postagem, evento, pergunta, local, grupo e missão. Atualmente estão disponíveis 154 serviços (78 requisições *post* e 76 requisições *get*), que

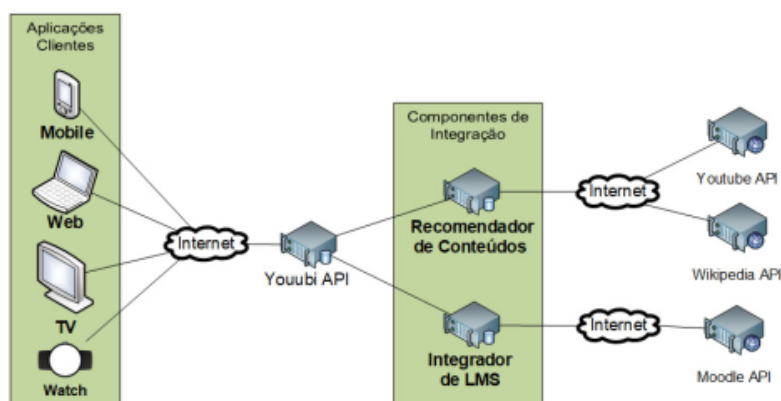


Figura 1. Visão geral do ambiente Edubi

possibilitam às aplicações clientes implementarem as seguintes funcionalidades: autoria de conteúdo (postagens, eventos, perguntas e locais); recomendação das EE; rede social; comentários; chat; grupo de discussão (por meio da entidade grupo); notificações; gamificação (pontos de experiência, medalhas e rankings); missões (sequência de objetivos); geolocalização; *bookmark* (lista de conteúdos salvos); monitoramento quantitativo de interações; e moderação.

Conforme pode ser observado na Figura 2, a arquitetura da Youubi API é composta por seis módulos principais: *Communication*: representa a camada de comunicação, pois permite que as aplicações clientes utilizem a Youubi API para requisitar os serviços do Youubi Server. O acesso a API é feito por controle de sessão, ou seja, os serviços da API requerem o ID do usuário e um *token* de sessão (disponibilizado após o login); *Manager*: é considerado o núcleo do servidor por desempenhar importantes funções: tratar as requisições passadas pela camada de comunicação, gerenciar as regras de domínio definidas na Youubi API, distribuir demandas aos outros módulos, entre outras responsabilidades; *Collector*: é responsável por coletar dados em outros sistemas que disponibilizam uma API de acesso. Os conteúdos coletados automaticamente por esse módulo, em plataformas externas, precisam ser validados por um moderador para que finalmente possam ficar visíveis para os demais usuários. *Recommender*: é responsável por gerar uma lista de objetos recomendados para os usuários, composta por todas as entidades elementares; *Model*: é responsável por representar o modelo de dados das entidades da arquitetura; e *Persistence*: é responsável pela persistência das informações no banco de dados (BD).

3. LESD-Onto: *Learning Environment Semantic Data mapping Ontology*

Para guiar o desenvolvimento da ontologia, utilizou-se a metodologia *Ontology Development 101* [Noy et al. 2004], que consiste de um processo iterativo de execução simples no desenvolvimento de ontologias, dividido em sete fases:

1. **Determinar o domínio e o escopo da ontologia:** esta fase consiste em responder “para quê”, “onde” e “como” a ontologia será empregada.
2. **Considerar o reuso de ontologias existentes:** nesta fase é realizada a busca por ontologias que tenham relação com a ontologia a ser desenvolvida.
3. **Enumerar termos importantes da ontologia:** esta fase tem como objetivo identificar um conjunto de termos frequentemente utilizados em um domínio de co-

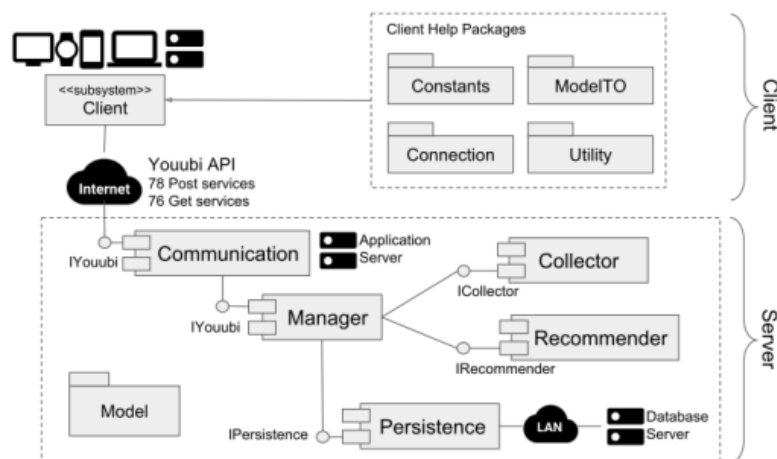


Figura 2. Arquitetura da Youubi API (adaptado de Monteiro)

nhecimento.

4. **Definir as classes e a hierarquia:** esta fase consiste em definir um conjunto de classes e organizá-las de forma hierárquica.
5. **Definir as propriedades das classes:** esta fase tem como objetivo dar sentido aos relacionamentos entre as classes.
6. **Definir as restrições:** nesta fase são definidas as restrições das propriedades das classes. Para propriedades que relacionem classes a dados, é necessário denominar os tipos de dado envolvidos.
7. **Criar indivíduos:** esta fase consiste no processo de criar indivíduos da ontologia a partir da hierarquia das classes em concordância com suas propriedades.

O domínio coberto pela LESD-Onto é referente a ambientes virtuais de aprendizagem, com o objetivo de auxiliar a integração de ambientes virtuais de aprendizagem, fornecendo o mapeamento de conceitos relacionados aos modelos de dados de um sistema para outro. Os ambientes selecionados para a construção da ontologia foram o Edubi e o Moodle. A ontologia pode ser utilizada por professores e alunos da modalidade de educação a distância, usuários desses ambientes. O responsável pela manutenção e expansão da LESD-Onto é um administrador do Edubi, que poderá, futuramente, incluir conceitos relacionados a outros ambientes de aprendizagem.

Uma das maneiras de determinar o escopo da ontologia é esboçar uma lista de questões de competências que uma ontologia deve ser capaz de responder. Essas questões servem como guia para identificar se a ontologia cumpre o seu objetivo [Grüninger and Fox 1995]. Visando mapear os dados providos pelo Moodle em entidades conhecidas pelo Youubi, as questões de competências que a ontologia LESD-Onto deve responder são:

1. Qual papel um administrador do Moodle pode cumprir no Youubi?
2. Qual papel um professor do Moodle pode cumprir no Youubi?
3. Qual papel um estudante do Moodle pode cumprir no Youubi?
4. A que corresponde um módulo do Moodle no modelo de dados Youubi?
5. A que corresponde um evento do Moodle no modelo de dados Youubi?

6. Que informações de usuário no Moodle podem ser atribuídas a um usuário Youubi?
7. Que informações de módulo Moodle podem ser atribuídas a um conteúdo Youubi?
8. Que informações de evento Moodle podem ser atribuídas a um conteúdo Youubi?

Algumas ontologias criadas para descrever conceitos relacionados ao Moodle podem ser encontradas na literatura [Piovesan et al. 2011, Yusof and Mansur 2013]. Apesar disso, nenhuma das ontologias encontradas descreve o modelo de dados do ambiente, de modo a ser útil para o mapeamento dos dados. Além disso, não foi encontrada nenhuma ontologia que descreva o modelo de dados do Youubi. Desse modo, não foi possível o reuso de ontologias.

Para identificar os termos que serviram de base para a construção da ontologia, foi realizado um *brainstorming* para identificar quais informações do Moodle poderiam ser úteis para o Youubi. Observando o conteúdo fornecido pela API do Moodle, foram identificados os seguintes termos: *Event, Module, Assignment, Book, Chat, Choice, Database, Feedback, File, Folder, Forum, Glossary, Lesson, Page, Quiz, Survey, URL, Wiki, Workshop, User, Guest, Student, Teacher, Administrator e Course Creator*. O Youubi, por sua vez, contém um modelo de dados mais enxuto. Avaliando a sua documentação, foi possível destacar os seguintes termos: *Person, Moderator, User, Content, Event, Location e Post*.

A concepção da ontologia foi auxiliada pela ferramenta Protégé, uma ferramenta que oferece suporte para a construção, teste e manutenção de ontologias e um *framework* para a construção de sistemas inteligentes.

Conforme apresentado na Figura 3, duas classes derivam da classe nativa *owl:Thing*, são elas: *Moodle_Data_Model* e *Youubi_Data_Model*. Essas classes são utilizadas para separar os modelos de dados de cada ambiente. Ao todo, foram definidas 32 classes, sendo 24 para descrever o modelo de dados do Moodle, e 8 para descrever o modelo de dados do Youubi. Além disso, é possível observar, na Figura 3, as relações de equivalência entre subclasses do *Moodle_Data_Model* e do *Youubi_Data_Model*.



Figura 3. Propriedades da LESD-Onto

O próximo passo foi definir as propriedades da ontologia. Ao todo, foram definidas 34 propriedades que relacionam classes a tipos de dados. As propriedades definidas para o Moodle estão descritas na Figura 4 (a) e para o Youubi na Figura 4 (b).

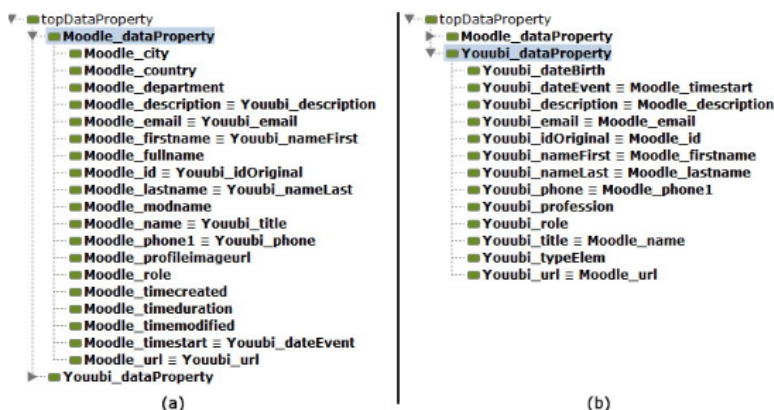


Figura 4. Propriedades da LESD-Onto

4. Validação

Alguns aspectos são fundamentais para serem validados na ontologia, antes que ela seja aplicada a uma situação prática, conforme apresentado em Gómez-Pérez (2003).

Inicialmente, deve-se avaliar se há erros de consistência lógica durante a execução do *reasoner*, verificando se não há possibilidade de obter conclusões contraditórias a partir de entradas válidas. Para isso, foram criados indivíduos no Moodle com características diversas. Com os indivíduos criados e após a execução do *reasoner*, foi analisado em que classe cada indivíduo foi inferido. A Figura 5 mostra indivíduos que têm papéis administrativos no Moodle, sendo inferidos como moderador no Youubi.

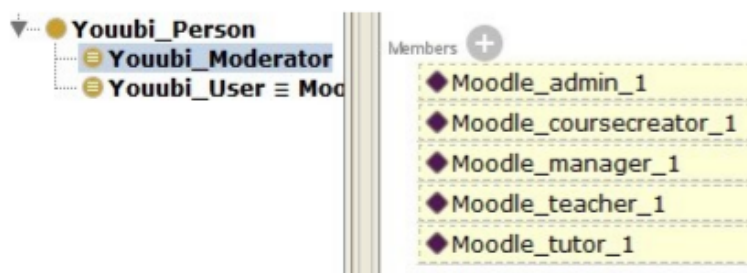


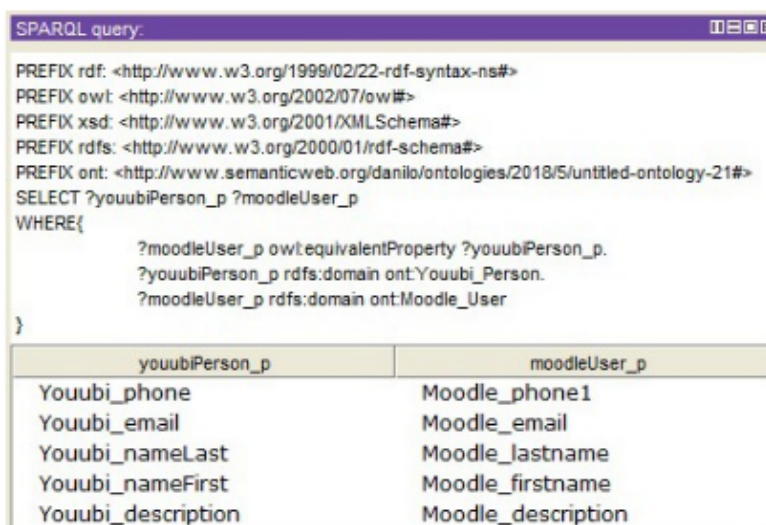
Figura 5. Inferência de usuários do Moodle em papéis do Youubi

Com os indivíduos inferidos, é possível responder às questões de competências. A Tabela 1, a seguir, apresenta as questões de competências definidas durante o processo de criação da ontologia, e as respectivas respostas fornecidas por meio da inferência.

Tabela 1. Respostas às questões de competências retornadas por inferência

Questão	Resposta	A resposta está correta?
Questão 1	Youubi_Moderator	Sim
Questão 2	Youubi_Moderator	Sim
Questão 3	Youubi_User	Sim
Questão 4	Youubi_Post	Sim
Questão 5	Youubi_Event	Sim

Os resultados obtidos não mostraram nenhuma inconsistência, ou seja, todos os indivíduos foram inseridos nas classes esperadas. No entanto, para responder às questões 6, 7 e 8, foi necessário realizar consultas por propriedades que sejam equivalentes em classes diferentes. Para isso, foram utilizadas consultas SparQL para verificar quais propriedades de uma classe no Moodle equivalem a propriedades de outra classe no Youubi. Com os resultados retornados é possível observar que a ontologia é capaz de identificar como representar uma classe do Moodle em uma classe do Youubi, confirmando que a ontologia cumpre com o seu papel. A Figura 6 ilustra como uma consulta SparQL responde à questão de competência 6.



```
SPARQL query:
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX ont: <http://www.semanticweb.org/danilo/ontologies/2018/5/untitled-ontology-21#>
SELECT ?youubiPerson_p ?moodleUser_p
WHERE{
    ?moodleUser_p owl:equivalentProperty ?youubiPerson_p.
    ?youubiPerson_p rdfs:domain ont:Youubi_Person.
    ?moodleUser_p rdfs:domain ont:Moodle_User
}
```

youubiPerson_p	moodleUser_p
Youubi_phone	Moodle_phone1
Youubi_email	Moodle_email
Youubi_nameLast	Moodle_lastname
Youubi_nameFirst	Moodle_firstname
Youubi_description	Moodle_description

Figura 6. Consulta SparQL e resultado da questão 6

Em seguida, foram avaliadas a completude e a concisão da ontologia. Nessa etapa, é verificado se tudo o que deveria estar na ontologia é explicitamente declarado, ou pode ser inferido, e se a ontologia não possui definições desnecessárias, inúteis ou redundantes. Para isso, pode ser utilizado um visualizador gráfico. Desse modo, é possível observar alguma classe ou propriedade que não possua relação com o restante da ontologia.

Com a finalidade de visualizar a ontologia graficamente e verificar se esta permanece hiper conectada com suas classes e propriedades, utilizou-se a ferramenta WebVOWL. De acordo com a Figura 7, é possível verificar que não há classes e propriedades que não se encontrem conectadas. Desse modo, garante-se que a ontologia é hiper conectada, o que significa dizer que as informações inseridas podem ser recuperadas por meio de consultas na ontologia, garantindo sua completude e concisão.

5. Considerações Finais e Trabalhos Futuros

O presente trabalho apresentou a LESD-Onto, uma ontologia para o mapeamento semântico de dados entre ambientes de aprendizagem. O desenvolvimento foi guiado pela metodologia *Ontology Development 101*. A ontologia é capaz de descrever o modelo de dados do ambiente de aprendizagem Moodle e da Youubi API, que fornece um conjunto de serviços para a construção de aplicações para computação ubíqua, que, no contexto deste trabalho, foi utilizada no ambiente Edubi. Vale ressaltar que é possível

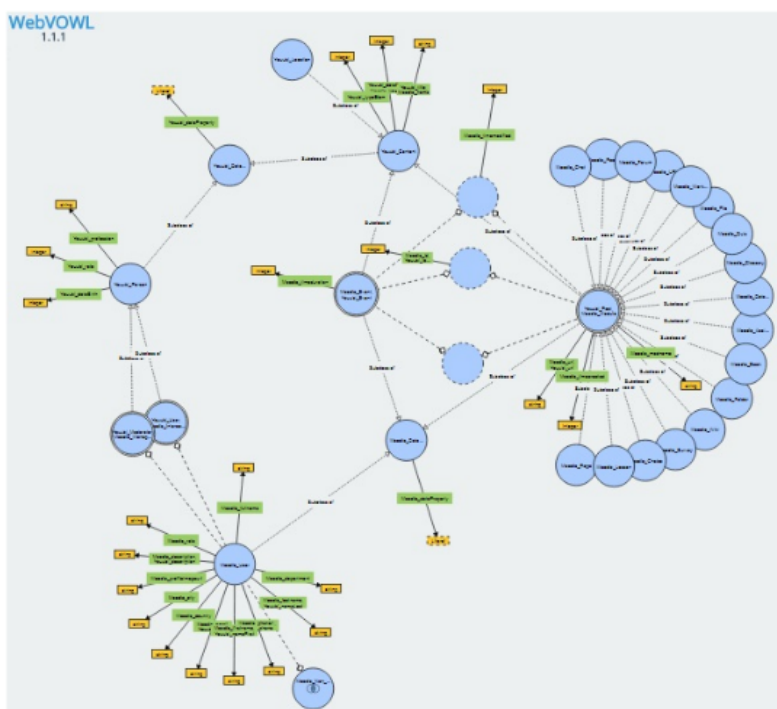


Figura 7. Visão da ontologia utilizando a WebVOL

expandir a ontologia para abarcar outros ambientes de aprendizagem usando um modelo de dados genérico que englobe todas as características desse ambiente.

Após a construção da ontologia, foram avaliados três aspectos: a consistência, a completude e a concisão. Com a validação, foi constatado que a ontologia responde às questões de competência, sem erros de execução e sem redundâncias. Esses resultados dão indícios da viabilidade da utilização da ontologia para realizar o mapeamento de dados entre sistemas heterogêneos.

Como trabalho futuro, sugere-se inserir a ontologia em um componente que possa consumir serviços de ambientes de aprendizagem, para realizar o mapeamento das informações e fornecê-las para a Youubi API. Assim, aplicações clientes do ambiente Edubi poderão fornecer novas funcionalidades para os usuários a partir dos dados coletados dos ambientes externos, não apenas o Moodle.

Referências

- Bezerra, S. F., Silva, S., Neto, F. M., Silva, P., and de Sousa Monteiro, B. (2017). Sistema de recomendação ubíquo integrando hiper mídias baseada em ontologia. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 28, page 1793.
- de Sousa Monteiro, B., de Oliveira, E. J., Gomes, A. S., and Neto, F. M. (2015). Youubi: Ambiente de aprendizagem ubíqua. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 26, page 111.

- Gómez-Pérez, A. (2004). Ontology evaluation. In *Handbook on ontologies*, pages 251–273. Springer.
- Gruber, T., Ontology, I. L. L., and Özsu, M. T. (2009). Encyclopedia of database systems. *Ontology*.
- Grüninger, M. and Fox, M. S. (1995). Methodology for the design and evaluation of ontologies.
- Hohpe, G. and Woolf, B. (2004). *Enterprise integration patterns: Designing, building, and deploying messaging solutions*. Addison-Wesley Professional.
- Noy, N. F., McGuinness, D. L., et al. (2004). Ontology development 101: A guide to creating your first ontology. 2001. See <http://protege.stanford.edu/publications>.
- Piovesan, S. D., do Amaral, É. M. H., Librelotto, G. R., and Medina, R. D. (2011). Ontologia para o ambiente virtual de aprendizagem moodle. *Cadernos de Informática*, 6(1):269–272.
- Romaní, C. C. and Moravec, J. W. (2011). *Aprendizaje invisible: Hacia una nueva ecología de la educación*, volume 3. Edicions Universitat Barcelona.
- Vinoski, S. (2002). Putting the “web” into web services. web services interaction models. 2. *IEEE Internet Computing*, 6(4):90–92.
- Yusof, N. and Mansur, A. B. F. (2013). Ontology development of e-learning moodle for social learning network analysis. *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, 7(6):853–858.